

OCF Resource to AllJoyn Interface Mapping Specification

VERSION 2.0.3 | June 2019



OPEN CONNECTIVITY
FOUNDATION™

CONTACT admin@openconnectivity.org

Copyright Open Connectivity Foundation, Inc. © 2019.
All Rights Reserved.

Legal Disclaimer

NOTHING CONTAINED IN THIS DOCUMENT SHALL BE DEEMED AS GRANTING YOU ANY KIND OF LICENSE IN ITS CONTENT, EITHER EXPRESSLY OR IMPLIEDLY, OR TO ANY INTELLECTUAL PROPERTY OWNED OR CONTROLLED BY ANY OF THE AUTHORS OR DEVELOPERS OF THIS DOCUMENT. THE INFORMATION CONTAINED HEREIN IS PROVIDED ON AN "AS IS" BASIS, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE AUTHORS AND DEVELOPERS OF THIS SPECIFICATION HEREBY DISCLAIM ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, STATUTORY OR AT COMMON LAW, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OPEN CONNECTIVITY FOUNDATION, INC. FURTHER DISCLAIMS ANY AND ALL WARRANTIES OF NON-INFRINGEMENT, ACCURACY OR LACK OF VIRUSES.

The OCF logo is a trademark of Open Connectivity Foundation, Inc. in the United States or other countries. *Other names and brands may be claimed as the property of others.

Copyright © 2017-2019 Open Connectivity Foundation, Inc. All rights reserved.

Copying or other form of reproduction and/or distribution of these works are strictly prohibited.

21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62

CONTENTS

1	Scope	1
2	Normative references	1
3	Terms and definitions	1
4	Document conventions and organization.....	1
4.1	Conventions.....	1
4.2	Notation	2
5	Theory of operation	2
5.1	Interworking approach	2
5.2	Mapping syntax.....	3
5.2.1	Introduction	3
5.2.2	General	3
5.2.3	Value assignment	3
5.2.4	Property naming	3
5.2.5	Arrays.....	3
5.2.6	Default mapping	3
5.2.7	Conditional mapping	3
5.2.8	Loops	3
5.2.9	Method invocation	4
6	Device type mapping	4
6.1	AllJoyn device types to OCF device types.....	4
6.2	OCF device types with no AllJoyn equivalent.....	5
7	Resource to interface equivalence.....	5
7.1	Introduction.....	5
7.2	Environment.CurrentAirQuality mapping	7
7.3	Environment.CurrentAirQualityLevel mapping	7
7.4	Operation.ClimateControlMode mapping	7
7.5	Operation.FanSpeedLevel mapping	7
7.6	Operation.HeatingZone mapping.....	7
7.7	Operation.OnOffStatus, Operation.OnControl, and Operation.OffControl mapping.....	7
7.8	Operation.OvenCyclePhase	8
8	Detailed mapping APIs	8
8.1	Introduction.....	8
8.2	Current Air Quality	8
8.2.1	Derived model	8
8.2.2	Property definition	8
8.2.3	Derived model definition	9
8.3	Current Air Quality Level.....	11
8.3.1	Derived model	11

63	8.3.2	Property definition	11
64	8.3.3	Derived model definition	12
65	8.4	Current Humidity	13
66	8.4.1	Derived model	13
67	8.4.2	Property definition	13
68	8.4.3	Derived model definition	13
69	8.5	Current Temperature	14
70	8.5.1	Derived model	14
71	8.5.2	Property definition	14
72	8.5.3	Derived model definition	14
73	8.6	Target Humidity	16
74	8.6.1	Derived model	16
75	8.6.2	Property definition	16
76	8.6.3	Derived model definition	16
77	8.7	Target Temperature	18
78	8.7.1	Derived model	18
79	8.7.2	Property definition	18
80	8.7.3	Derived model definition	18
81	8.8	Audio Volume	20
82	8.8.1	Derived model	20
83	8.8.2	Property definition	20
84	8.8.3	Derived model definition	20
85	8.9	Climate Control Mode	21
86	8.9.1	Derived model	21
87	8.9.2	Property definition	21
88	8.9.3	Derived model definition	22
89	8.10	Closed Status	23
90	8.10.1	Derived model	23
91	8.10.2	Property definition	23
92	8.10.3	Derived model definition	23
93	8.11	Cycle Control	24
94	8.11.1	Derived model	24
95	8.11.2	Property definition	24
96	8.11.3	Derived model definition	25
97	8.12	Fan Speed Level	26
98	8.12.1	Derived model	26
99	8.12.2	Property definition	26
100	8.12.3	Derived model definition	26
101	8.13	Heating Zone	27
102	8.13.1	Derived model	27
103	8.13.2	Property definition	27
104	8.13.3	Derived model definition	28
105	8.14	HVAC Fan Mode	29
106	8.14.1	Derived model	29

107	8.14.2	Property definition	29
108	8.14.3	Derived model definition	30
109	8.15	On/Off Control	31
110	8.15.1	Derived model	31
111	8.15.2	Property definition	31
112	8.15.3	Derived model definition	31
113	8.16	On Off Mapping	32
114	8.16.1	Derived model	32
115	8.16.2	Property definition	32
116	8.16.3	Derived model definition	32
117	8.17	Oven Cycle Phase	33
118	8.17.1	Derived model	33
119	8.17.2	Property definition	33
120	8.17.3	Derived model definition	34
121			

122

123

No table of figures entries found.

Figures

Tables

124	
125	Table 1 – AllJoyn to OCF device type mapping4
126	Table 2 – OCF device types with no AllJoyn equivalent.....5
127	Table 3 – AllJoyn interface to OCF resource type mapping – minimum interface set6
128	Table 4 – AllJoyn interface to OCF resource type mapping – optional interface set.....6
129	Table 5 – Interface to resource summary8
130	Table 6 – The property mapping for "asa.environment.currentairquality"9
131	Table 7 – The properties of "asa.environment.currentairquality"9
132	Table 8 – The property mapping for "asa.environment.currentairqualitylevel" 11
133	Table 9 – The properties of "asa.environment.currentairqualitylevel". 11
134	Table 10 – The property mapping for "asa.environment.currenthumidity" 13
135	Table 11 – The properties of "asa.environment.currenthumidity" 13
136	Table 12 – The property mapping for "asa.environment.currenttemperature" 14
137	Table 13 – The properties of "asa.environment.currenttemperature" 14
138	Table 14 – The property mapping for "asa.environment.targethumidity" 16
139	Table 15 – The properties of "asa.environment.targethumidity" 16
140	Table 16 – The property mapping for "asa.environment.targettemperature" 18
141	Table 17 – The properties of "asa.environment.targettemperature" 18
142	Table 18 – The property mapping for "asa.operation.audiovolume"20
143	Table 19 – The properties of "asa.operation.audiovolume"20
144	Table 20 – The property mapping for "asa.operation.climatecontrolmode"21
145	Table 21 – The properties of "asa.operation.climatecontrolmode"22
146	Table 22 – The property mapping for "asa.operation.closedstatus"23
147	Table 23 – The properties of "asa.operation.closedstatus"23
148	Table 24 – The property mapping for "asa.operation.cyclecontrol"24
149	Table 25 – The properties of "asa.operation.cyclecontrol"24
150	Table 26 – The property mapping for "asa.operation.fanspeedlevel"26
151	Table 27 – The properties of "asa.operation.fanspeedlevel"26
152	Table 28 – The property mapping for "asa.operation.heatingzone"27
153	Table 29 – The properties of "asa.operation.heatingzone"28
154	Table 30 – The property mapping for "asa.operation.hvacfanmode"29
155	Table 31 – The properties of "asa.operation.hvacfanmode"30
156	Table 32 – The property mapping for "asa.operation.offcontrol"31
157	Table 33 – The properties of "asa.operation.offcontrol"31
158	Table 34 – The property mapping for "asa.operation.oncontrol"31
159	Table 35 – The properties of "asa.operation.oncontrol"31
160	Table 36 – The property mapping for "asa.operation.onoffstatus"32
161	Table 37 – The properties of "asa.operation.onoffstatus"32
162	Table 38 – The property mapping for "asa.operation.ovencyclephase"33

163 Table 39 – The properties of "asa.operation.ovencyclephase".33
164

165 **1 Scope**

166 This document provides detailed mapping information to provide equivalency between AllJoyn
167 defined Interfaces and OCF defined Resources.

168 This document provides mapping for Device Types (AllJoyn to/from OCF), identifies equivalent
169 OCF Resources for both mandatory and optional AllJoyn interfaces and for each interface defines
170 the detailed Property by Property mapping using OCF defined extensions to JSON schema to
171 programmatically define the mappings.

172 **1 Normative references**

173 The following documents are referred to in the text in such a way that some or all of their content
174 constitutes requirements of this document. For dated references, only the edition cited applies. For
175 undated references, the latest edition of the referenced document (including any amendments)
176 applies.

177 ISO/IEC 30118-1:2018 Information technology -- Open Connectivity Foundation (OCF)
178 Specification -- Part 1: Core specification
179 <https://www.iso.org/standard/53238.html>
180 Latest version available at: https://openconnectivity.org/specs/OCF_Core_Specification.pdf

181 ISO/IEC 30118-4:2018 Information technology -- Open Connectivity Foundation (OCF)
182 Specification -- Part 4: Resource type specification
183 <https://www.iso.org/standard/74241.html>
184 Latest version available at:
185 https://openconnectivity.org/specs/OCF_Resource_Type_Specification.pdf

186 ISO/IEC 30118-5:2019, Information technology – Open Connectivity Foundation (OCF)
187 Specification – Part 5: Smart home device specification
188 <https://www.iso.org/standard/74242.html>
189 Latest version available at: https://openconnectivity.org/specs/OCF_Device_Specification.pdf

190 Derived Models for Interoperability between IoT Ecosystems, Stevens & Merriam, March 2016
191 [https://www.iab.org/wp-content/IAB-uploads/2016/03/OCF-Derived-Models-for-Interoperability-](https://www.iab.org/wp-content/IAB-uploads/2016/03/OCF-Derived-Models-for-Interoperability-Between-IoT-Ecosystems_v2-examples.pdf)
192 [Between-IoT-Ecosystems_v2-examples.pdf](https://www.iab.org/wp-content/IAB-uploads/2016/03/OCF-Derived-Models-for-Interoperability-Between-IoT-Ecosystems_v2-examples.pdf)

193 AllJoyn Common Data Model Interface Definitions
194 <https://wiki.alljoyn.org/cdm>

195 **2 Terms and definitions**

196 For the purposes of this document, the terms and definitions given in ISO/IEC 30118-1:2018 and
197 the following apply.

198 ISO and IEC maintain terminological databases for use in standardization at the following
199 addresses:

200 – ISO Online browsing platform: available at <https://www.iso.org/obp>

201 – IEC Electropedia: available at <http://www.electropedia.org/>

202 **3 Document conventions and organization**

203 **3.1 Conventions**

204 In this document a number of terms, conditions, mechanisms, sequences, parameters, events,
205 states, or similar terms are printed with the first letter of each word in uppercase and the rest

206 lowercase (e.g., Network Architecture). Any lowercase uses of these words have the normal
207 technical English meaning.

208 **3.2 Notation**

209 In this document, features are described as required, recommended, allowed or DEPRECATED as
210 follows:

211 Required (or shall or mandatory).

212 These basic features shall be implemented to comply with the Mapping Specification. The
213 phrases "shall not", and "PROHIBITED" indicate behaviour that is prohibited, i.e. that if
214 performed means the implementation is not in compliance.

215 Recommended (or should).

216 These features add functionality supported by the Mapping Specification and should be
217 implemented. Recommended features take advantage of the capabilities the Mapping
218 Specification, usually without imposing major increase of complexity. Notice that for compliance
219 testing, if a recommended feature is implemented, it shall meet the specified requirements to
220 be in compliance with these guidelines. Some recommended features could become
221 requirements in the future. The phrase "should not" indicates behaviour that is permitted but
222 not recommended.

223 Allowed (or allowed).

224 These features are neither required nor recommended by the Mapping Specification, but if the
225 feature is implemented, it shall meet the specified requirements to be in compliance with these
226 guidelines.

227 Conditionally allowed (CA)

228 The definition or behaviour depends on a condition. If the specified condition is met, then the
229 definition or behaviour is allowed, otherwise it is not allowed.

230 Conditionally required (CR)

231 The definition or behaviour depends on a condition. If the specified condition is met, then the
232 definition or behaviour is required. Otherwise the definition or behaviour is allowed as default
233 unless specifically defined as not allowed.

234 DEPRECATED

235 Although these features are still described in this document, they should not be implemented
236 except for backward compatibility. The occurrence of a deprecated feature during operation of
237 an implementation compliant with the current document has no effect on the implementation's
238 operation and does not produce any error conditions. Backward compatibility may require that
239 a feature is implemented and functions as specified but it shall never be used by
240 implementations compliant with this document.

241 Strings that are to be taken literally are enclosed in "double quotes".

242 Words that are emphasized are printed in *italic*.

243 **4 Theory of operation**

244 **4.1 Interworking approach**

245 The interworking between AllJoyn defined interfaces and OCF defined Resource Types is modelled
246 using the derived model syntax described in Derived Models for Interoperability between IoT
247 Ecosystems. Determination of the minimum set of AllJoyn interfaces for which equivalency is
248 required within the OCF data model was done by listing the set of interfaces required for each of

249 the device types defined by the CDM Project inside of AllJoyn. Where the AllJoyn interface
250 supports methods then an actuation design pattern is applied.

251 **4.2 Mapping syntax**

252 **4.2.1 Introduction**

253 Within the defined syntax for derived modelling used by this document there are two blocks that
254 define the actual Property-Property equivalence or mapping. These blocks are identified by the
255 keywords "x-to-ocf" and "x-from-ocf". Derived Models for Interoperability between IoT Ecosystems
256 does not define a rigid syntax for these blocks; they are free form string arrays that contain pseudo-
257 coded mapping logic.

258 Within this document we apply the rules defined in clause 5.2 to these blocks to ensure consistency
259 and re-usability and extensibility of the mapping logic that is defined.

260 **4.2.2 General**

261 All statements are terminated with a carriage return.

262 **4.2.3 Value assignment**

263 The equals sign (=) is used to assign one value to another. The assignee is on the left of the
264 operator; the value being assigned on the right.

265 **4.2.4 Property naming**

266 All Property names are identical to the name used by the original model; for example, from the
267 OCF Temperature Resource the Property name "temperature" is used whereas when referred to
268 the derived ecosystem then the semantically equivalent Property name is used.

269 When the same name is used by both OCF and the derived ecosystem for semantically equivalent
270 values then the name of the OCF defined Property is prepended by the ecosystem designator "ocf"
271 to avoid ambiguity (e.g. "ocf.step").

272 **4.2.5 Arrays**

273 An array element is indicated by the use of square brackets "[]" with the index of the element
274 contained therein, e.g. range[1]. All arrays start at an index of 0. If an entire array is being
275 referenced then no index is included, e.g. selectablehumiditylevels[].

276 **4.2.6 Default mapping**

277 There are cases where the specified mapping is not possible as one or more of the Properties
278 being mapped is optional in the source model. In all such instances a default mapping is provided.
279 The default map is indicated by the prepending of an "otherwise:" modifier to the assignment. (e.g.
280 "otherwise: step = 1").

281 **4.2.7 Conditional mapping**

282 When a mapping is dependent on the meeting of other conditions then the syntax:

283 if "condition", "mapping".

284 Is applied.

285 E.g. if step >0, ocf.step = step.

286 **4.2.8 Loops**

287 When a mapping can be represented by a repeated loop governed by some condition then the
288 syntax:

289 for "initialize", "condition", "increment": "mapping"

290 Where:

291 "initialize" is an initial local loop control variable setting.

292 "condition" is the loop controller, the loop repeats until the condition evaluates to "false".

293 "increment" allows for update of the control variable, if omitted an increment of "1" is assumed.

294 Is applied.

295 E.g. for $x=0, x < \text{sizeof}(\text{supportedmodes})$: `ocf.supportedmodes[x] = modearray[supportedmodes[x]]`

296 **4.2.9 Method invocation**

297 The invocation of a method or remote procedure call (RPC) from the derived ecosystem as part of
298 the mapping from an OCF Resource is indicated by the use of a double colon "::" delimiter between
299 the applicable resource, service, interface or other construct identifier and the method or RPC
300 name. The method name always includes trailing parentheses which would include any parameters
301 should they be passed.

302 For example, when dealing with the `switchon()` method from AllJoyn this gives a complete method
303 invocation as: `operation.oncontrol::switchon()`.

304 **5 Device type mapping**

305 **5.1 AllJoyn device types to OCF device types**

306 Table 1 captures the equivalency mapping between AllJoyn defined Device Types (see AllJoyn
307 Common Data Model Interface Definitions) and OCF defined Device Types (see Table 10-1 in
308 ISO/IEC 30118-5:2019). The minimum interface set for the AllJoyn definitions is provided in the
309 HAE Theory of Operation; the minimum Resource sets for each OCF Device is provided in
310 ISO/IEC 30118-5:2019.

311 **Table 1 – AllJoyn to OCF device type mapping**

Classification	AllJoyn Device Type	AllJoyn ID	OCF Device Type
Air Care	Air Conditioner	5	oic.d.airconditioner
	Air Purifier	9	oic.d.airpurifier
	Air Quality Monitor	11	oic.d.airqualitymonitor
	Dehumidifier	8	oic.d.dehumidifier
	Humidifier	7	oic.d.humidifier
	Electric Fan	10	oic.d.fan
	Thermostat	6	oic.d.thermostat
Fabric Care	Clothes Washer	12	oic.d.washer
	Clothes Dryer	13	oic.d.dryer
	Clothes Washer-Dryer	14	oic.d.washerdryer
Food Preservation	Refrigerator	2	oic.d.refrigerator
	Ice-Maker	4	oic.r.icemaker (maps to Resource)
	Freezer	3	oic.d.freezer
Food Preparation	Oven	17	oic.d.oven

	Cooktop	18	oic.d.cooktop
	Cookerhood	19	oic.d.cookerhood
	Food probe	20	oic.d.foodprobe
Dish Care	Dishwasher	15	oic.d.dishwasher
Floor Care	Robot Cleaner	16	oic.d.robotcleaner
Entertainment	Television	21	oic.d.tv
	Set Top Box (STB)	22	oic.d.stb

312 **5.2 OCF device types with no AllJoyn equivalent**

313 Table 2 captures the Device Types defined by OCF have no direct equivalent in AllJoyn, they shall
314 all be mapped to an AllJoyn Device Type of "Other" (Id of "1").

315 **Table 2 – OCF device types with no AllJoyn equivalent**

OCF Device Name	OCF Device Type
Receiver	oic.d.receiver
Blind	oic.d.blind
Door	oic.d.door
Garage Door	oic.d.garagedoor
Generic Sensor	oic.d.sensor
Light	oic.d.light
Smart Plug	oic.d.smartplug
Switch	oic.d.switch
Water Valve	oic.d.watervalve
Printer	oic.d.printer
Multi-Function Printer	oic.d.multifunctionprinter
Scanner	oic.r.scanner
Camera	oic.d.camera
Security Panel	oic.d.securitypanel
Smart Lock	oic.d.smartlock

316 **6 Resource to interface equivalence**

317 **6.1 Introduction**

318 Clause 7 captures the equivalency mapping between AllJoyn defined Interfaces (see AllJoyn
319 Common Data Model Interface Definitions) and OCF defined Resource Types (see ISO/IEC 30118-
320 4:2018). Detailed Property by Property mappings are provided in clause 8.

321 Table 3 captures the mappings for Interfaces that are part of the minimum set for an AllJoyn Device.

322 Table 4 captures the mappings for Interfaces that are optional for an AllJoyn Device; deep
323 translation for these interfaces via derived modelling is not within the scope of this release of the
324 document.

Table 3 – AllJoyn interface to OCF resource type mapping – minimum interface set

AllJoyn Interface	OCF Resource Type Name	OCF Resource Type ID	OCF Interface(s)
Environment.CurrentAirQuality	Air Quality Collection	oic.r.airqualitycollection	oic.if.s
Environment.CurrentAirQualityLevel	Air Quality Collection	oic.r.airqualitycollection	oic.if.s
Environment.CurrentHumidity	Humidity	oic.r.humidity	oic.if.s
Environment.CurrentTemperature	Temperature	oic.r.temperature	oic.if.s
Environment.TargetHumidity	Humidity	oic.r.humidity, oic.r.selectablelevels	oic.if.a
Environment.TargetTemperature	Temperature	oic.r.temperature	oic.if.a
Operation.AudioVolume	Audio Controls	oic.r.audio	oic.if.a
Operation.Channel	Not mapped		
Operation.ClimateControlMode	Mode	oic.r.mode	oic.if.a
	Operational State	oic.r.operational.state	oic.if.s
Operation.ClosedStatus	Door	oic.r.door	oic.if.s
Operation.CycleControl	Operational State	oic.r.operational.state	oic.if.s
Operation.FanSpeedLevel	Air Flow	oic.r.airflow	oic.if.a
Operation.HeatingZone	Heating Zone Collection	oic.r.heatingzonecollection	oic.if.s
Operation.HvacFanMode	Mode	oic.r.mode	oic.if.a
Operation.OnOffStatus	Binary Switch	oic.r.switch.binary	oic.if.s
Operation.OvenCyclePhase	Operational State	oic.r.operationalstate	oic.if.s

Table 4 – AllJoyn interface to OCF resource type mapping – optional interface set

AllJoyn Interface	OCF Resource Type Name	OCF Resource Type ID	OCF Interface(s)
Environment.TargetTemperatureLevel	Mode	oic.r.mode	oic.if.a
Environment.WaterLevel	TBD	TBD	oic.if.s
Environment.WindDirection	Air Flow	oic.r.airflow	oic.if.a
Operation.AirRecirculationMode	Mode	oic.r.mode	oic.if.a
Operation.Alerts	TBD	TBD	TBD
Operation.AudioVideoInput	Media Source List	oic.r.media.input	oic.if.a
Operation.BatteryStatus	Battery	oic.r.energy.battery	oic.if.s
Operation.CurrentPower	Energy Usage	oic.r.energy.usage	oic.if.s
Operation.DishWashingCyclePhase	Operational State	oic.r.operationalstate	oic.if.s
Operation.EnergyUsage	Energy Usage	oic.r.energy.usage	oic.if.s
Operation.FilterStatus	TBD	TBD	TBD
Operation.LaundryCyclePhase	Mode	oic.r.mode	oic.if.s
Operation.MoistureOutputLevel	Mode	oic.r.mode	oic.if.a
Operation.PlugInUnits	TBD	TBD	TBD

Operation.RapidMode	Refrigeration	oic.r.refrigeration	oic.if.a
Operation.RemoteControllability	TBD	TBD	TBD
Operation.RepeatMode	Ecomode	oic.r.ecomode	oic.if.a
Operation.ResourceSaving	TBD	TBD	TBD
Operation.RobotCleaningCyclePhase	Mode	oic.r.mode	oic.if.s
Operation.SoilLevel	Mode	oic.r.mode	oic.if.a
Operation.SpinSpeedLevel	Mode	oic.r.mode	oic.if.a
Operation.Timer	Time Period	oic.r.time.period	oic.if.s

328 **6.2 Environment.CurrentAirQuality mapping**

329 If more than one instance of the AirQuality interface is exposed, then each instance maps to an
330 instance of the OCF AirQuality Resource. The mapping defined in clause 8.2 describes the
331 population of the OCF AirQuality Resource. Even if there is only a single instance of an OCF
332 AirQuality Resource this shall be included in an instance of an OCF AirQualityCollection. The
333 number of links in the collection equates to the number of instances of the AllJoyn CurrentAirQuality
334 interface that are exposed. When mapping from OCF the valueType of the Resource shall be
335 introspected, this API is invoked only if this is set to "Measured".

336 **6.3 Environment.CurrentAirQualityLevel mapping**

337 If more than one instance of the AirQualityLevel interface is exposed, then each instance maps to
338 an instance of the OCF AirQuality Resource. The mapping defined in clause 8.2 describes the
339 population of the OCF AirQuality Resource. Even if there is only a single instance of an OCF
340 AirQuality Resource then this shall be included in an instance of an OCF AirQualityCollection. The
341 number of links in the collection equates to the number of instances of the AllJoyn CurrentAirQuality
342 interface that are exposed. When mapping from OCF the valueType of the Resource shall be
343 introspected, this API is invoked only if this is set to "Qualitative".

344 **6.4 Operation.ClimateControlMode mapping**

345 ClimateControlMode has three Properties; these map as follows: mode and supportedmodes maps
346 to the Mode Resource, operationalstate maps to the OperationalState Resource This can be
347 represented in OCF either as two distinct Resource instances or a single instance with two
348 Resource Types (oic.r.mode, oic.r.operationalstate).

349 **6.5 Operation.FanSpeedLevel mapping**

350 The setting of the FanSpeedLevel to "0x00" (off) is handled via the "OffControl" interface rather
351 than writing directly to this interface. In such a case an instance of Binary Switch shall be exposed
352 on the OCF side; this can be modelled as AirFlowControl which is then a collection of Binary Switch
353 and AirFlow.

354 **6.6 Operation.HeatingZone mapping**

355 Each element in the array of heating zones within the AllJoyn HeatingZone interface maps to an
356 instance of OCF HeatingZone, itself a link in an instance of an OCF HeatingZoneCollection. The
357 mapping defined clause 8.13 describes the population of the OCF HeatingZone Resource that
358 constitutes the Resources that are contained in the collection.

359 **6.7 Operation.OnOffStatus, Operation.OnControl, and Operation.OffControl mapping**

360 A discovered instance of a Binary Switch is always mapped to an Operation.OnOffStatus interface.
361 A RETRIEVE on a Binary Switch maps to an action on an instance of an Operation.OnOffStatus
362 Interface. An UPDATE on a Binary Switch maps to a method invocation on either
363 Operation.OnControl or OffControl. value = true maps to Operation.OnControl value = false
364 maps to Operation.OffControl.

365 **6.8 Operation.OvenCyclePhase**

366 OvenCyclePhase cyclephase Property pre-defines values 0x00-0x7F, 0x80-0xFF is for vendor
367 specific values. The mapping defined in clause 8 covers only specification defined values. Any
368 vendor defined value shall be represented in OCF using the x.<organization> syntax for a vendor
369 defined Property.

370 **7 Detailed mapping APIs**

371 **7.1 Introduction**

372 This clause provides a mapping description (using JSON that aligns with the Derived Modelling
373 syntax described in Derived Models for Interoperability between IoT Ecosystems for all Interfaces
374 and Resources that are within scope.

375 The derived model definitions presented in clause 8 are formatted for readability, and so may
376 appear to have extra line breaks.

377 Table 5 provides a reference and link to the per Interface clauses.

378

Table 5 – Interface to resource summary

AllJoyn Interface Name	Equivalent Resource(s)	Clause
Environment.CurrentAirQuality	oic.r.airqualitycollection	8.2
Environment.CurrentAirQualityLevel	oic.r.airqualitycollection	8.3
Environment.CurrentHumidity	oic.r.humidity	8.4
Environment.CurrentTemperature	oic.r.temperature	8.5
Environment.TargetHumidity	oic.r.humidity, oic.r.selectablelevels	8.6
Environment.TargetTemperature	oic.r.temperature	8.7
Operation.AudioVolume	oic.r.audio	8.8
Operation.ClimateControlMode	oic.r.mode, oic.r.operationalstate	8.9
Operation.ClosedStatus	oic.r.door	8.10
Operation.CycleControl	oic.r.operational.state	8.11
Operation.FanSpeedLevel	oic.r.airflow	8.12
Operation.HeatingZone	oic.r.heatingzonecollection	8.13
Operation.HvacFanMode	oic.r.mode	8.14
Operation.OnControl, Operation.OffControl	oic.r.switch.binary	8.15
Operation.OnOffStatus,	oic.r.switch.binary	8.16
Operation.OvenCyclePhase	oic.r.operationalstate	8.17

379 **7.2 Current Air Quality**

380 **7.2.1 Derived model**

381 The derived model: "asa.environment.currentairquality".

382 **7.2.2 Property definition**

383 Table 6 provides the detailed per Property mapping for "asa.environment.currentairquality".

Table 6 – The property mapping for "asa.environment.currentairquality".

AllJoyn Property name	OCF Resource	To OCF	From OCF
minvalue	oic.r.airquality	range[0] = minvalue	minvalue = range[0]
maxvalue	oic.r.airquality	range[1] = maxvalue	maxvalue = range[1]
contaminanttype	oic.r.airquality	valuetype = Measuredcontaminanttypearray [CH2O,CO2,CO,PM2_5,PM10,VOC] ocf.contaminanttype = contaminanttypearray[contaminanttype]	contaminanttype = indexof contaminanttypearray[ocf.contaminanttype]
currentvalue	oic.r.airquality	contaminantvalue = currentvalue	currentvalue = contaminantvalue
updatemintime	oic.r.value.conditional	ocf.minnotifyperiod = updatemintime	updatemintime = ocf.minnotifyperiod
precision	oic.r.airquality	ocf.precision = precision	precision = ocf.precision

385 Table 7 provides the details of the Properties that are part of "asa.environment.currentairquality".

386

Table 7 – The properties of "asa.environment.currentairquality".

AllJoyn Property name	Property	Type	Required	Description
minvalue		number	yes	
maxvalue		number	yes	
contaminanttype		integer	yes	The contaminant type
currentvalue		number	yes	
updatemintime		integer	yes	
precision		number	yes	

387 7.2.3 Derived model definition

```

388 {
389   "id": "http://openinterconnect.org/asamapping/schemas/asa.environment.currentairquality.json#",
390   "$schema": "http://json-schema.org/draft-04/schema#",
391   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
392   "title": "Current Air Quality",
393   "definitions": {
394     "asa.environment.currentairquality": {
395       "type": "object",
396       "properties": {
397         "contaminanttype": {
398           "type": "integer",
399           "description": "The contaminant type",
400           "x-ocf-conversion": {
401             "x-ocf-alias": "oic.r.airquality",
402             "x-to-ocf": [
403               "valuetype = Measured",
404               "contaminanttypearray = [CH2O,CO2,CO,PM2_5,PM10,VOC]",
405               "ocf.contaminanttype = contaminanttypearray[contaminanttype]"
406             ],
407             "x-from-ocf": [
408               "contaminanttype = indexof contaminanttypearray[ocf.contaminanttype]"
409             ]
410           }
411         },
412         "currentvalue": {
413           "type": "number",

```

```

414         "x-ocf-conversion": {
415             "x-ocf-alias": "oic.r.airquality",
416             "x-to-ocf": [
417                 "contaminantvalue = currentvalue"
418             ],
419             "x-from-ocf": [
420                 "currentvalue = contaminantvalue"
421             ]
422         },
423     },
424     "minvalue": {
425         "type": "number",
426         "x-ocf-conversion": {
427             "x-ocf-alias": "oic.r.airquality",
428             "x-to-ocf": [
429                 "range[0] = minvalue"
430             ],
431             "x-from-ocf": [
432                 "minvalue = range[0]"
433             ]
434         },
435     },
436     "maxvalue": {
437         "type": "number",
438         "x-ocf-conversion": {
439             "x-ocf-alias": "oic.r.airquality",
440             "x-to-ocf": [
441                 "range[1] = maxvalue"
442             ],
443             "x-from-ocf": [
444                 "maxvalue = range[1]"
445             ]
446         },
447     },
448     "precision": {
449         "type": "number",
450         "x-ocf-conversion": {
451             "x-ocf-alias": "oic.r.airquality",
452             "x-to-ocf": [
453                 "ocf.precision = precision"
454             ],
455             "x-from-ocf": [
456                 "precision = ocf.precision"
457             ]
458         },
459     },
460     "updatemintime": {
461         "type": "integer",
462         "x-ocf-conversion": {
463             "x-ocf-alias": "oic.r.value.conditional",
464             "x-to-ocf": [
465                 "ocf.minnotifyperiod = updatemintime"
466             ],
467             "x-from-ocf": [
468                 "updatemintime = ocf.minnotifyperiod"
469             ]
470         },
471     },
472 },
473 },
474 },
475 "type": "object",
476 "allOf": [
477     {"$ref": "#/definitions/asa.environment.currentairquality"}
478 ],
479 "required": ["contaminanttype", "currentvalue", "minvalue", "maxvalue", "precision", "updatemintime"]
480 }
481

```

482 **7.3 Current Air Quality Level**

483 **7.3.1 Derived model**

484 The derived model: "asa.environment.currentairqualitylevel".

485 **7.3.2 Property definition**

486 Table 8 provides the detailed per Property mapping for "asa.environment.currentairqualitylevel".

487 **Table 8 – The property mapping for "asa.environment.currentairqualitylevel".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
contaminanttype		oic.r.airquality	valuetype = Qualitativeif contaminanttype = 0, ocf.contaminanttype = CH2Oif contaminanttype = 1, ocf.contaminanttype = CO2if contaminanttype = 2, ocf.contaminanttype = COif contaminanttype = 3, ocf.contaminanttype = PM2_5if contaminanttype = 4, ocf.contaminanttype = PM10if contaminanttype = 5, ocf.contaminanttype = VOCif contaminanttype = 253, ocf.contaminanttype = Smokeif contaminanttype = 254, ocf.contaminanttype = Odorif contaminanttype = 255, ocf.contaminanttype = AirPollution	if ocf.contaminanttype = CH2O, contaminanttype = 0if ocf.contaminanttype = CO2, contaminanttype = 1if ocf.contaminanttype = CO, contaminanttype = 2if ocf.contaminanttype = PM2_5, contaminanttype = 3if ocf.contaminanttype = PM10, contaminanttype = 4if ocf.contaminanttype = VOC, contaminanttype = 5if ocf.contaminanttype = Smoke, contaminanttype = 253if ocf.contaminanttype = Odor, contaminanttype = 254if ocf.contaminanttype = AirPollution, contaminanttype = 255
maxlevel		oic.r.airquality	range[0] = 0range[1] = maxvalue	maxvalue = range[1]
currentlevel		oic.r.airquality	contaminantvalue = currentlevel	currentlevel = contaminantvalue

488 Table 9 provides the details of the Properties that are part of
489 "asa.environment.currentairqualitylevel".

490 **Table 9 – The properties of "asa.environment.currentairqualitylevel".**

AllJoyn name	Property	Type	Required	Description
contaminanttype		integer	yes	The contaminant type

maxlevel	integer	yes	
currentlevel	integer	yes	

491 7.3.3 Derived model definition

```

492 {
493   "id":
494   "http://openinterconnect.org/asamapping/schemas/asa.environment.currentairqualitylevel.json#",
495   "$schema": "http://json-schema.org/draft-04/schema#",
496   "description" : "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
497   "title": "Current Air Quality Level",
498   "definitions": {
499     "asa.environment.currentairqualitylevel": {
500       "type": "object",
501       "properties": {
502         "contaminanttype": {
503           "type": "integer",
504           "description": "The contaminant type",
505           "x-ocf-conversion": {
506             "x-ocf-alias": "oic.r.airquality",
507             "x-to-ocf": [
508               "valuetype = Qualitative",
509               "if contaminanttype = 0, ocf.contaminanttype = CH2O",
510               "if contaminanttype = 1, ocf.contaminanttype = CO2",
511               "if contaminanttype = 2, ocf.contaminanttype = CO",
512               "if contaminanttype = 3, ocf.contaminanttype = PM2_5",
513               "if contaminanttype = 4, ocf.contaminanttype = PM10",
514               "if contaminanttype = 5, ocf.contaminanttype = VOC",
515               "if contaminanttype = 253, ocf.contaminanttype = Smoke",
516               "if contaminanttype = 254, ocf.contaminanttype = Odor",
517               "if contaminanttype = 255, ocf.contaminanttype = AirPollution"
518             ],
519             "x-from-ocf": [
520               "if ocf.contaminanttype = CH2O, contaminanttype = 0",
521               "if ocf.contaminanttype = CO2, contaminanttype = 1",
522               "if ocf.contaminanttype = CO, contaminanttype = 2",
523               "if ocf.contaminanttype = PM2_5, contaminanttype = 3",
524               "if ocf.contaminanttype = PM10, contaminanttype = 4",
525               "if ocf.contaminanttype = VOC, contaminanttype = 5",
526               "if ocf.contaminanttype = Smoke, contaminanttype = 253",
527               "if ocf.contaminanttype = Odor, contaminanttype = 254",
528               "if ocf.contaminanttype = AirPollution, contaminanttype = 255"
529             ]
530           }
531         },
532         "currentlevel": {
533           "type": "integer",
534           "x-ocf-conversion": {
535             "x-ocf-alias": "oic.r.airquality",
536             "x-to-ocf": [
537               "contaminantvalue = currentlevel"
538             ],
539             "x-from-ocf": [
540               "currentlevel = contaminantvalue"
541             ]
542           }
543         },
544         "maxlevel": {
545           "type": "integer",
546           "x-ocf-conversion": {
547             "x-ocf-alias": "oic.r.airquality",
548             "x-to-ocf": [
549               "range[0] = 0",
550               "range[1] = maxvalue"
551             ],
552             "x-from-ocf": [
553               "maxvalue = range[1]"
554             ]
555           }
556         }
557       }
558     }
559   }
560 }

```

```

558     }
559   },
560   "type": "object",
561   "allOf": [
562     { "$ref": "#/definitions/asa.environment.currentairqualitylevel" }
563   ],
564   "required": ["contaminanttype", "currentlevel", "maxlevel"]
565 }
566

```

567 7.4 Current Humidity

568 7.4.1 Derived model

569 The derived model: "asa.environment.currenthumidity".

570 7.4.2 Property definition

571 Table 10 provides the detailed per Property mapping for "asa.environment.currenthumidity".

572 **Table 10 – The property mapping for "asa.environment.currenthumidity".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
maxvalue		oic.r.humidity	range[0] = Orange[1] = maxvalue	maxvalue = range[1]
currentvalue		oic.r.humidity	humidity = currentValue	currentvalue = humidity

573 Table 11 provides the details of the Properties that are part of "asa.environment.currenthumidity".

574 **Table 11 – The properties of "asa.environment.currenthumidity".**

AllJoyn name	Property	Type	Required	Description
maxvalue		number	yes	Max measured value for humidty
currentvalue		number	yes	Measured value

575 7.4.3 Derived model definition

```

576 {
577   "id": "http://openinterconnect.org/asamapping/schemas/asa.environment.currenthumidity.json#",
578   "$schema": "http://json-schema.org/draft-04/schema#",
579   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
580   "title": "Current Humidity",
581   "definitions": {
582     "asa.environment.currenthumidity": {
583       "type": "object",
584       "properties": {
585         "currentvalue": {
586           "type": "number",
587           "description": "Measured value",
588           "x-ocf-conversion": {
589             "x-ocf-alias": "oic.r.humidity",
590             "x-to-ocf": [
591               "humidity = currentValue"
592             ],
593             "x-from-ocf": [
594               "currentvalue = humidity"
595             ]
596           }
597         },
598         "maxvalue": {
599           "type": "number",
600           "description": "Max measured value for humidty",
601           "x-ocf-conversion": {
602             "x-ocf-alias": "oic.r.humidity",

```

```

603         "x-to-ocf": [
604             "range[0] = 0",
605             "range[1] = maxvalue"
606         ],
607         "x-from-ocf": [
608             "maxvalue = range[1]"
609         ]
610     }
611 }
612 }
613 }
614 },
615 "type": "object",
616 "allOf": [
617     {"$ref": "#/definitions/asa.environment.currenthumidity"}
618 ],
619 "required": [ "currentvalue", "maxvalue" ]
620 }
621

```

622 7.5 Current Temperature

623 7.5.1 Derived model

624 The derived model: "asa.environment.currenttemperature".

625 7.5.2 Property definition

626 Table 12 provides the detailed per Property mapping for "asa.environment.currenttemperature".

627 **Table 12 – The property mapping for "asa.environment.currenttemperature".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
precision		oic.r.temperature	ocf.precision = precision	precision = ocf.precision
currentvalue		oic.r.temperature	temperature = currentValueunits = C	oneOf
updatemintime		oic.r.value.conditional	ocf.minnotifyperiod = updatemintime	updatemintime = ocf.minnotifyperiod

628 Table 13 provides the details of the Properties that are part of
629 "asa.environment.currenttemperature".

630 **Table 13 – The properties of "asa.environment.currenttemperature".**

AllJoyn name	Property	Type	Required	Description
precision		number	yes	
currentvalue		number	yes	Measured value
updatemintime		integer	yes	

631 7.5.3 Derived model definition

```

632 {
633     "id": "http://openinterconnect.org/asamapping/schemas/asa.environment.currenttemperature.json#",
634     "$schema": "http://json-schema.org/draft-04/schema#",
635     "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
636     "title": "Current Temperature",
637     "definitions": {
638         "asa.environment.currenttemperature": {
639             "type": "object",
640             "properties": {
641                 "currentvalue": {
642                     "type": "number",
643                     "description": "Measured value",
644                     "x-ocf-conversion": {

```

```

645     "x-ocf-alias": "oic.r.temperature",
646     "x-to-ocf": [
647         "temperature = currentValue",
648         "units = C"
649     ],
650     "x-from-ocf": {
651         "oneOf": [
652             {
653                 "properties": {
654                     "units": "string",
655                     "enum": ["C"]
656                 },
657                 "x-from-ocf": [
658                     "currentvalue = temperature"
659                 ]
660             },
661             {
662                 "properties": {
663                     "units": "string",
664                     "enum": ["F"]
665                 },
666                 "x-from-ocf": [
667                     "currentvalue = (temperature-32)*5/9"
668                 ]
669             },
670             {
671                 "properties": {
672                     "units": "string",
673                     "enum": ["K"]
674                 },
675                 "x-from-ocf": [
676                     "currentvalue = temperature-273.15"
677                 ]
678             }
679         ]
680     }
681 },
682 "precision": {
683     "type": "number",
684     "x-ocf-conversion": {
685         "x-ocf-alias": "oic.r.temperature",
686         "x-to-ocf": [
687             "ocf.precision = precision"
688         ],
689         "x-from-ocf": [
690             "precision = ocf.precision"
691         ]
692     }
693 },
694 "updatemintime": {
695     "type": "integer",
696     "x-ocf-conversion": {
697         "x-ocf-alias": "oic.r.value.conditional",
698         "x-to-ocf": [
699             "ocf.minnotifyperiod = updatemintime"
700         ],
701         "x-from-ocf": [
702             "updatemintime = ocf.minnotifyperiod"
703         ]
704     }
705 }
706 }
707 }
708 }
709 },
710 "type": "object",
711 "allOf": [
712     {"$ref": "#/definitions/asa.environment.currenttemperature"}
713 ],
714 "required": [ "currentvalue", "precision", "updatemintime" ]

```

715 }
716

717 **7.6 Target Humidity**

718 **7.6.1 Derived model**

719 The derived model: "asa.environment.targethumidity".

720 **7.6.2 Property definition**

721 Table 14 provides the detailed per Property mapping for "asa.environment.targethumidity".

722 **Table 14 – The property mapping for "asa.environment.targethumidity".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
minvalue	oic.r.humidity	range[0] = minvalue	minvalue = range[0]otherwise: minvalue = 0
targetvalue	oic.r.humidity,oic.r.selectablelevels	if minvalue != maxvalue, ocf.desiredhumidity = targetvalue;ocf.targetlevel = selectablehumiditylevels[0].if minvalue == maxvalue, ocf.targetlevel = targetvalue.	if x-ocf-alias == oic.r.humidity, targetvalue = desiredhumidity.if x-ocf-alias == oic.r.selectablelevels, targetvalue = targetlevel.
maxvalue	oic.r.humidity	range[1] = maxvalue	maxvalue = range[1]otherwise: maxvalue = 100
stepvalue	oic.r.humidity	step = stepvalue	stepvalue = stepotherwise: stepvalue = 1
selectablehumiditylevels	oic.r.selectablelevels	availablelevels[] = selectablehumiditylevels[]	selectablehumiditylevels[] = availablelevels[]

723 Table 15 provides the details of the Properties that are part of "asa.environment.targethumidity".

724 **Table 15 – The properties of "asa.environment.targethumidity".**

AllJoyn Property name	Type	Required	Description
minvalue	number	yes	
targetvalue	number	yes	Measured value
maxvalue	number	yes	
stepvalue	number	yes	
selectablehumiditylevels	array	yes	

725 **7.6.3 Derived model definition**

```
726 {
727   "id": "http://openinterconnect.org/asamapping/schemas/asa.environment.targethumidity.json#",
728   "$schema": "http://json-schema.org/draft-04/schema#",
729   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
730   "title": "Target Humidity",
731   "definitions": {
732     "asa.environment.targethumidity": {
733       "type": "object",
734       "properties": {
735         "targetvalue": {
```



```

736         "type": "number",
737         "description": "Measured value",
738         "x-ocf-conversion": {
739             "x-ocf-alias": "oic.r.humidity,oic.r.selectablelevels",
740             "x-to-ocf": [
741                 "if minvalue != maxvalue, ocf.desiredhumidity = targetvalue;ocf.targetlevel =
742 selectablehumiditylevels[0].",
743                 "if minvalue == maxvalue, ocf.targetlevel = targetvalue."
744             ],
745             "x-from-ocf": [
746                 "if x-ocf-alias == oic.r.humidity, targetvalue = desiredhumidity.",
747                 "if x-ocf-alias == oic.r.selectablelevels, targetvalue = targetlevel."
748             ]
749         }
750     },
751     "minvalue": {
752         "type": "number",
753         "x-ocf-conversion": {
754             "x-ocf-alias": "oic.r.humidity",
755             "x-to-ocf": [
756                 "range[0] = minvalue"
757             ],
758             "x-from-ocf": [
759                 "minvalue = range[0]",
760                 "otherwise: minvalue = 0"
761             ]
762         }
763     },
764     "maxvalue": {
765         "type": "number",
766         "x-ocf-conversion": {
767             "x-ocf-alias": "oic.r.humidity",
768             "x-to-ocf": [
769                 "range[1] = maxvalue"
770             ],
771             "x-from-ocf": [
772                 "maxvalue = range[1]",
773                 "otherwise: maxvalue = 100"
774             ]
775         }
776     },
777     "stepvalue": {
778         "type": "number",
779         "x-ocf-conversion": {
780             "x-ocf-alias": "oic.r.humidity",
781             "x-to-ocf": [
782                 "step = stepvalue"
783             ],
784             "x-from-ocf": [
785                 "stepvalue = step",
786                 "otherwise: stepvalue = 1"
787             ]
788         }
789     },
790     "selectablehumiditylevels": {
791         "type": "array",
792         "items": {
793             "type": "number"
794         },
795         "x-ocf-conversion": {
796             "x-ocf-alias": "oic.r.selectablelevels",
797             "x-to-ocf": [
798                 "availablelevels[] = selectablehumiditylevels[]"
799             ],
800             "x-from-ocf": [
801                 "selectablehumiditylevels[] = availablelevels[]"
802             ]
803         }
804     }
805 }
806 }

```

```

807     },
808     "type": "object",
809     "allOf": [
810       { "$ref": "#/definitions/asa.environment.targethumidity" }
811     ],
812     "required": [ "targetvalue", "minvalue", "maxvalue", "stepvalue", "selectablehumiditylevels" ]
813   }
814 }

```

815 7.7 Target Temperature

816 7.7.1 Derived model

817 The derived model: "asa.environment.targettemperature".

818 7.7.2 Property definition

819 Table 16 provides the detailed per Property mapping for "asa.environment.targettemperature".

820 **Table 16 – The property mapping for "asa.environment.targettemperature".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
minvalue		oic.r.temperature	range[0] = minvalue	minvalue = range[0] otherwise: minvalue = -MAXINT
targetvalue		oic.r.temperature	temperature = targetvalue units = C	oneOf
maxvalue		oic.r.temperature	range[1] = maxvalue	maxvalue = range[1] otherwise: maxvalue = MAXINT
step		oic.r.temperature	ocf.step = step	step = ocf.step otherwise: step = undefined (0x00)

821 Table 17 provides the details of the Properties that are part of "asa.environment.targettemperature".

822 **Table 17 – The properties of "asa.environment.targettemperature".**

AllJoyn name	Property	Type	Required	Description
minvalue		number	yes	
targetvalue		number	yes	Measured value
maxvalue		number	yes	
step		number	yes	

823 7.7.3 Derived model definition

```

824 {
825   "id": "http://openinterconnect.org/asamapping/schemas/asa.environment.targettemperature.json#",
826   "$schema": "http://json-schema.org/draft-04/schema#",
827   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
828   "title": "Target Temperature",
829   "definitions": {
830     "asa.environment.targettemperature": {
831       "type": "object",
832       "properties": {
833         "targetvalue": {
834           "type": "number",
835           "description": "Measured value",
836           "x-ocf-conversion": {
837             "x-ocf-alias": "oic.r.temperature",
838             "x-to-ocf": [
839               "temperature = targetvalue",
840               "units = C"

```

```

841     ],
842     "x-from-ocf": {
843         "oneOf": [
844             {
845                 "properties": {
846                     "units": "string",
847                     "enum": ["C"]
848                 },
849                 "x-from-ocf": [
850                     "targetvalue = temperature"
851                 ]
852             },
853             {
854                 "properties": {
855                     "units": "string",
856                     "enum": ["F"]
857                 },
858                 "x-from-ocf": [
859                     "targetvalue = (temperature-32)*5/9"
860                 ]
861             },
862             {
863                 "properties": {
864                     "units": "string",
865                     "enum": ["K"]
866                 },
867                 "x-from-ocf": [
868                     "targetvalue = temperature-273.15"
869                 ]
870             }
871         ]
872     }
873 },
874 "minvalue": {
875     "type": "number",
876     "x-ocf-conversion": {
877         "x-ocf-alias": "oic.r.temperature",
878         "x-to-ocf": [
879             "range[0] = minvalue"
880         ],
881     },
882     "x-from-ocf": [
883         "minvalue = range[0]",
884         "otherwise: minvalue = -MAXINT"
885     ]
886 },
887 "maxvalue": {
888     "type": "number",
889     "x-ocf-conversion": {
890         "x-ocf-alias": "oic.r.temperature",
891         "x-to-ocf": [
892             "range[1] = maxvalue"
893         ],
894     },
895     "x-from-ocf": [
896         "maxvalue = range[1]",
897         "otherwise: maxvalue = MAXINT"
898     ]
899 },
900 "step": {
901     "type": "number",
902     "x-ocf-conversion": {
903         "x-ocf-alias": "oic.r.temperature",
904         "x-to-ocf": [
905             "ocf.step = step"
906         ],
907     },
908     "x-from-ocf": [
909         "step = ocf.step",
910         "otherwise: step = undefined (0x00)"
911     ]

```

```

912     }
913   }
914 }
915 },
916 },
917 "type": "object",
918 "allOf": [
919   {"$ref": "#/definitions/asa.environment.targettemperature"}
920 ],
921 "required": [ "targetvalue", "minvalue", "maxvalue", "step" ]
922 }
923

```

924 7.8 Audio Volume

925 7.8.1 Derived model

926 The derived model: "asa.operation.audiovolume".

927 7.8.2 Property definition

928 Table 18 provides the detailed per Property mapping for "asa.operation.audiovolume".

929 **Table 18 – The property mapping for "asa.operation.audiovolume".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
mute		oic.r.audio	ocf.mute = mute	mute = ocf.mute
maxvolume		oic.r.audio	range[0] = 0range[1] = maxvolume	maxvolume = range[1]otherwise: maxvalue = 100
volume		oic.r.audio	ocf.volume = volume	volume = ocf.volume

930 Table 19 provides the details of the Properties that are part of "asa.operation.audiovolume".

931 **Table 19 – The properties of "asa.operation.audiovolume".**

AllJoyn name	Property	Type	Required	Description
mute		boolean	yes	
maxvolume		integer	yes	
volume		integer	yes	Speaker volume index

932 7.8.3 Derived model definition

```

933 {
934   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.audiovolume.json#",
935   "$schema": "http://json-schema.org/draft-04/schema#",
936   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
937   "title": "Audio Volume",
938   "definitions": {
939     "asa.operation.audiovolume": {
940       "type": "object",
941       "properties": {
942         "volume": {
943           "type": "integer",
944           "description": "Speaker volume index",
945           "x-ocf-conversion": {
946             "x-ocf-alias": "oic.r.audio",
947             "x-to-ocf": [
948               "ocf.volume = volume"
949             ],
950             "x-from-ocf": [
951               "volume = ocf.volume"
952             ]
953           }
954         },
955         "maxvolume": {

```

```

956     "type": "integer",
957     "x-ocf-conversion": {
958       "x-ocf-alias": "oic.r.audio",
959       "x-to-ocf": [
960         "range[0] = 0",
961         "range[1] = maxvolume"
962       ],
963       "x-from-ocf": [
964         "maxvolume = range[1]",
965         "otherwise: maxvalue = 100"
966       ]
967     }
968   },
969   "mute": {
970     "type": "boolean",
971     "x-ocf-conversion": {
972       "x-ocf-alias": "oic.r.audio",
973       "x-to-ocf": [
974         "ocf.mute = mute"
975       ],
976       "x-from-ocf": [
977         "mute = ocf.mute"
978       ]
979     }
980   }
981 }
982 }
983 },
984 "type": "object",
985 "allOf": [
986   {"$ref": "#/definitions/asa.operation.audiovolume"}
987 ],
988 "required": [ "volume", "maxvolume", "mute" ]
989 }
990

```

991 7.9 Climate Control Mode

992 7.9.1 Derived model

993 The derived model: "asa.operation.climatecontrolmode".

994 7.9.2 Property definition

995 Table 20 provides the detailed per Property mapping for "asa.operation.climatecontrolmode".

996 **Table 20 – The property mapping for "asa.operation.climatecontrolmode".**

AllJoy n Proper ty name	OCF Resourc e	To OCF	From OCF
operati onalsta te	oic.r.ope rationalst ate	machinestates = [Idle,Heating,Cooling,PendingHeat,P endingCool,AuxilliaryHeat]currentma chinestate = machinestates[operationalstate]	statearray = [Idle,Heating,Cooling,PendingHeat ,PendingCool,AuxilliaryHeat]opera tionalstate = indexof statearray[currentmachinestate[0]]
suppor tedmo des	oic.r.mod e	modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,D ry,ContinuousDry]for x=0, x < sizeof(supportedmodes): ocf.supportedmodes[x] = modearray[supportedmodes[x]]	modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat, Dry,ContinuousDry]for x=0, x < sizeof(supportedmodes): supportedmodes[x] = indexof modearray[ocf.supportedmodes[x]]
mode	oic.r.mod e	modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,D	modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,

		ry,ContinuousDry]ocf.mode[0] = modearray[mode]	=	Dry,ContinuousDry]mode = indexof modeArray[ocf.mode[0]]
--	--	--	---	---

997 Table 21 provides the details of the Properties that are part of "asa.operation.climatecontrolmode".

998 **Table 21 – The properties of "asa.operation.climatecontrolmode".**

AllJoyn name	Property	Type	Required	Description
	operationalstate	integer	yes	Current status of device
	supportedmodes	array	yes	Array of supported modes
	mode	integer	yes	Current mode of device.

999 **7.9.3 Derived model definition**

```

1000 {
1001   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.climatecontrolmode.json#",
1002   "$schema": "http://json-schema.org/draft-04/schema#",
1003   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1004   "title": "Climate Control Mode",
1005   "definitions": {
1006     "asa.operation.climatecontrolmode": {
1007       "type": "object",
1008       "properties": {
1009         "mode": {
1010           "type": "integer",
1011           "description": "Current mode of device.",
1012           "x-ocf-conversion": {
1013             "x-ocf-alias": "oic.r.mode",
1014             "x-to-ocf": [
1015               "modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,Dry,ContinuousDry]",
1016               "ocf.mode[0] = modearray[mode]"
1017             ],
1018             "x-from-ocf": [
1019               "modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,Dry,ContinuousDry]",
1020               "mode = indexof modeArray[ocf.mode[0]]"
1021             ]
1022           }
1023         },
1024         "supportedmodes": {
1025           "type": "array",
1026           "items": {
1027             "type": "integer"
1028           },
1029           "description": "Array of supported modes",
1030           "x-ocf-conversion": {
1031             "x-ocf-alias": "oic.r.mode",
1032             "x-to-ocf": [
1033               "modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,Dry,ContinuousDry]",
1034               "for x=0, x < sizeof(supportedmodes): ocf.supportedmodes[x] =
1035 modearray[supportedmodes[x]]"
1036             ],
1037             "x-from-ocf": [
1038               "modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,Dry,ContinuousDry]",
1039               "for x=0, x < sizeof(supportedmodes): supportedmodes[x] = indexof
1040 modearray[ocf.supportedmodes[x]]"
1041             ]
1042           }
1043         },
1044         "operationalstate": {
1045           "type": "integer",
1046           "description": "Current status of device",
1047           "x-ocf-conversion": {
1048             "x-ocf-alias": "oic.r.operationalstate",
1049             "x-to-ocf": [

```

```

1050         "machinestates = [Idle,Heating,Cooling,PendingHeat,PendingCool,AuxilliaryHeat]",
1051         "currentmachinestate = machinestates[operationalstate]"
1052     ],
1053     "x-from-ocf": [
1054         "statearray = [Idle,Heating,Cooling,PendingHeat,PendingCool,AuxilliaryHeat]",
1055         "operationalstate = indexof statearray[currentmachinestate[0]]"
1056     ]
1057 }
1058 }
1059 }
1060 }
1061 },
1062 "type": "object",
1063 "allOf": [
1064     {"$ref": "#/definitions/asa.operation.climatecontrolmode"}
1065 ],
1066 "required": [ "mode","supportedmodes","operationalstate" ]
1067 }
1068

```

1069 **7.10 Closed Status**

1070 **7.10.1 Derived model**

1071 The derived model: "asa.operation.closedstatus".

1072 **7.10.2 Property definition**

1073 Table 22 provides the detailed per Property mapping for "asa.operation.closedstatus".

1074 **Table 22 – The property mapping for "asa.operation.closedstatus".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
isclosed		oic.r.door	if isClosed ocf.openState = Closed.if !isClosed ocf.openState = Open.	isClosed = (openState == Closed)

1075 Table 23 provides the details of the Properties that are part of "asa.operation.closedstatus".

1076 **Table 23 – The properties of "asa.operation.closedstatus".**

AllJoyn name	Property	Type	Required	Description
isclosed		boolean	yes	Open/Closed status Indicator

1077 **7.10.3 Derived model definition**

```

1078 {
1079     "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.closedstatus.json#",
1080     "$schema": "http://json-schema.org/draft-04/schema#",
1081     "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1082     "title": "Closed Status",
1083     "definitions": {
1084         "asa.operation.closedstatus": {
1085             "type": "object",
1086             "properties": {
1087                 "isclosed": {
1088                     "type": "boolean",
1089                     "description": "Open/Closed status Indicator",
1090                     "x-ocf-conversion": {
1091                         "x-ocf-alias": "oic.r.door",
1092                         "x-to-ocf": [
1093                             "if isClosed ocf.openState = Closed.",
1094                             "if !isClosed ocf.openState = Open."

```

```

1095     ],
1096     "x-from-ocf": [
1097       "isClosed = (openState == Closed)"
1098     ]
1099   }
1100 }
1101 }
1102 }
1103 },
1104 "type": "object",
1105 "allOf": [
1106   {"$ref": "#/definitions/asa.operation.closedstatus"}
1107 ],
1108 "required": [ "isclosed" ]
1109 }
1110

```

1111 **7.11 Cycle Control**

1112 **7.11.1 Derived model**

1113 The derived model: "asa.operation.cyclecontrol".

1114 **7.11.2 Property definition**

1115 Table 24 provides the detailed per Property mapping for "asa.operation.cyclecontrol".

1116 **Table 24 – The property mapping for "asa.operation.cyclecontrol".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
operationalstate	oic.r.operationalstate	statearray [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]currentmachinestate = statearray[operationalstate]	statearray [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]operationalstate = indexof statearray[currentmachinestate[0]]
executeoperationalcomand	oic.r.action		
SupportedOperationalcomands	oic.r.action		
supportedoperationalstates	oic.r.operationalstate	statearray [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]for x=0, x < sizeof(supportedoperationalstates): machinestates[x] = statearray[supportedoperationalstates[x]]	statearray [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]for x=0, x < sizeof(machinestates): supportedoperationalstates[x] = indexof statearray[machinestates[x]]

1117 Table 25 provides the details of the Properties that are part of "asa.operation.cyclecontrol".

1118 **Table 25 – The properties of "asa.operation.cyclecontrol".**

AllJoyn Property name	Type	Required	Description
operationalstate	integer	yes	Current operational state of the appliance
executeoperationalcomand		no	Execute an operational command

SupportedOperationalcommands	array	no	Array of operatinal commands supported by the appliance
supportedoperationalstates	array	yes	Array of operational states supported by the Appliance.

1119 7.11.3 Derived model definition

```

1120 {
1121   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.cyclecontrol.json#",
1122   "$schema": "http://json-schema.org/draft-04/schema#",
1123   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1124   "title": "Cycle Control",
1125   "definitions": {
1126     "asa.operation.cyclecontrol": {
1127       "type": "object",
1128       "properties": {
1129         "operationalstate": {
1130           "type": "integer",
1131           "description": "Current operational state of the appliance",
1132           "x-ocf-conversion": {
1133             "x-ocf-alias": "oic.r.operationalstate",
1134             "x-to-ocf": [
1135               "statearray = [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]",
1136               "currentmachinestate = statearray[operationalstate]"
1137             ],
1138             "x-from-ocf": [
1139               "statearray = [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]",
1140               "operationalstate = indexof statearray[currentmachinestate[0]]"
1141             ]
1142           }
1143         },
1144         "supportedoperationalstates": {
1145           "type": "array",
1146           "items": {
1147             "type": "integer"
1148           },
1149           "description": "Array of operational states supported by the Appliance.",
1150           "x-ocf-conversion": {
1151             "x-ocf-alias": "oic.r.operationalstate",
1152             "x-to-ocf": [
1153               "statearray = [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]",
1154               "for x=0, x < sizeof(supportedoperationalstates): machinestates[x] =
1155 statearray[supportedoperationalstates[x]]"
1156             ],
1157             "x-from-ocf": [
1158               "statearray = [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]",
1159               "for x=0, x < sizeof(machinestates): supportedoperationalstates[x] = indexof
1160 statearray[machinestates[x]]"
1161             ]
1162           }
1163         },
1164         "SupportedOperationalcommands": {
1165           "type": "array",
1166           "items": {
1167             "type": "integer"
1168           },
1169           "description": "Array of operatinal commands supported by the appliance",
1170           "x-ocf-conversion": {
1171             "x-ocf-alias": "oic.r.action"
1172           }
1173         },
1174         "executeoperationalcomand": {
1175           "x-ocf-type": "method",
1176           "description": "Execute an operational command",
1177           "x-ocf-conversion": {

```

```

1178         "x-ocf-alias": "oic.r.action"
1179     }
1180 }
1181 }
1182 }
1183 },
1184 "type": "object",
1185 "allOf": [
1186     {"$ref": "#/definitions/asa.operation.cyclecontrol"}
1187 ],
1188 "required": [ "operationalstate","supportedoperationalstates" ]
1189 }
1190

```

1191 **7.12 Fan Speed Level**

1192 **7.12.1 Derived model**

1193 The derived model: "asa.operation.fanspeedlevel".

1194 **7.12.2 Property definition**

1195 Table 26 provides the detailed per Property mapping for "asa.operation.fanspeedlevel".

1196 **Table 26 – The property mapping for "asa.operation.fanspeedlevel".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
fanspeedlevel		oic.r.airflow	speed = fanspeedlevel	fanspeedlevel = speed
maxfanspeedlevel		oic.r.airflow	range[0] = 0range[1] = maxfanspeedlevel	maxfanspeedlevel = range[1]otherwise: maxfanspeedlevel = 100
automode		oic.r.airflow	if automode != NotSupported(0xFF) ocf.automode = automodeelse no mapping	automode = ocf.automodeotherwise: automode = NotSupported(0xFF)

1197 Table 27 provides the details of the Properties that are part of "asa.operation.fanspeedlevel".

1198 **Table 27 – The properties of "asa.operation.fanspeedlevel".**

AllJoyn name	Property	Type	Required	Description
fanspeedlevel		integer	yes	Fan speed level. 0 = off.
maxfanspeedlevel		integer	yes	Max level allowed for fan speed
automode		integer	yes	Auto mode status.

1199 **7.12.3 Derived model definition**

```

1200 {
1201     "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.fanspeedlevel.json#",
1202     "$schema": "http://json-schema.org/draft-04/schema#",
1203     "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1204     "title": "Fan Speed Level",
1205     "definitions": {
1206         "asa.operation.fanspeedlevel": {
1207             "type": "object",
1208             "properties": {
1209                 "fanspeedlevel": {
1210                     "type": "integer",
1211                     "description": "Fan speed level. 0 = off.",

```

```

1212     "x-ocf-conversion": {
1213         "x-ocf-alias": "oic.r.airflow",
1214         "x-to-ocf": [
1215             "speed = fanspeedlevel"
1216         ],
1217         "x-from-ocf": [
1218             "fanspeedlevel = speed"
1219         ]
1220     },
1221 },
1222 "maxfanspeedlevel": {
1223     "type": "integer",
1224     "description": "Max level allowed for fan speed",
1225     "x-ocf-conversion": {
1226         "x-ocf-alias": "oic.r.airflow",
1227         "x-to-ocf": [
1228             "range[0] = 0",
1229             "range[1] = maxfanspeedlevel"
1230         ],
1231         "x-from-ocf": [
1232             "maxfanspeedlevel = range[1]",
1233             "otherwise: maxfanspeedlevel = 100"
1234         ]
1235     },
1236 },
1237 "automode": {
1238     "type": "integer",
1239     "description": "Auto mode status.",
1240     "x-ocf-conversion": {
1241         "x-ocf-alias": "oic.r.airflow",
1242         "x-to-ocf": [
1243             "if automode != NotSupported(0xFF)",
1244             " ocf.automode = automode",
1245             "else no mapping"
1246         ],
1247         "x-from-ocf": [
1248             "automode = ocf.automode",
1249             "otherwise: automode = NotSupported(0xFF)"
1250         ]
1251     },
1252 },
1253 },
1254 },
1255 },
1256 "type": "object",
1257 "allOf": [
1258     {"$ref": "#/definitions/asa.operation.fanspeedlevel"}
1259 ],
1260 "required": [ "fanspeedlevel", "maxfanspeedlevel", "automode" ]
1261 }
1262

```

1263 7.13 Heating Zone

1264 7.13.1 Derived model

1265 The derived model: "asa.operation.heatingzone".

1266 7.13.2 Property definition

1267 Table 28 provides the detailed per Property mapping for "asa.operation.heatingzone".

1268 **Table 28 – The property mapping for "asa.operation.heatingzone".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
numberofheatingzones	oic.r.heatingzonecollection	number of links in the collection = numberofheatingzones	numberofheatingzones = number of links in the collection

heatinglevels	oic.r.heatingzone	Instance of oic.r.heatingzone per array item for x=0, x<sizeof(heatinglevels): ocf.heatinglevel = maxheatinglevels[x]	for x=0;x<numlinks(oic.r.heatingzonecollection): heatinglevels[x] = ocf.heatinglevel
maxheatinglevels	oic.r.heatingzone	Instance of oic.r.heatingzone per array item for x=0, x<sizeof(maxheatinglevels): ocf.maxheatinglevel = maxheatinglevels[x]	for x=0;x<numlinks(oic.r.heatingzonecollection): maxheatinglevels[x] = ocf.maxheatinglevel

1269 Table 29 provides the details of the Properties that are part of "asa.operation.heatingzone".

1270 **Table 29 – The properties of "asa.operation.heatingzone".**

AllJoyn name	Property	Type	Required	Description
numberofheatingzones		integer	yes	Number of heating zones.
heatinglevels		array	yes	Current heating levels for each zone.
maxheatinglevels		array	yes	Max heating levels for each zone

1271 **7.13.3 Derived model definition**

```

1272 {
1273   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.heatingzone.json#",
1274   "$schema": "http://json-schema.org/draft-04/schema#",
1275   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1276   "title": "Heating Zone",
1277   "definitions": {
1278     "asa.operation.heatingzone": {
1279       "type": "object",
1280       "properties": {
1281         "numberofheatingzones": {
1282           "type": "integer",
1283           "description": "Number of heating zones.",
1284           "x-ocf-conversion": {
1285             "x-ocf-alias": "oic.r.heatingzonecollection",
1286             "x-to-ocf": [
1287               "number of links in the collection = numberofheatingzones"
1288             ],
1289             "x-from-ocf": [
1290               "numberofheatingzones = number of links in the collection"
1291             ]
1292           }
1293         },
1294         "maxheatinglevels": {
1295           "type": "array",
1296           "items": {
1297             "type": "integer"
1298           },
1299           "description": "Max heating levels for each zone",
1300           "x-ocf-conversion": {
1301             "x-ocf-alias": "oic.r.heatingzone",
1302             "x-to-ocf": [

```

```

1303         "Instance of oic.r.heatingzone per array item ",
1304         "for x=0, x<sizeof(maxheatinglevels): ocf.maxheatinglevel = maxheatinglevels[x]"
1305     ],
1306     "x-from-ocf": [
1307         "for x=0;x<numlinks(oic.r.heatingzonecollection): maxheatinglevels[x] =
1308 ocf.maxheatinglevel"
1309     ]
1310 }
1311 },
1312 "heatinglevels": {
1313     "type": "array",
1314     "items": {
1315         "type": "integer"
1316     },
1317     "description": "Current heating levels for each zone.",
1318     "x-ocf-conversion": {
1319         "x-ocf-alias": "oic.r.heatingzone",
1320         "x-to-ocf": [
1321             "Instance of oic.r.heatingzone per array item ",
1322             "for x=0, x<sizeof(heatinglevels): ocf.heatinglevel = maxheatinglevels[x]"
1323         ],
1324         "x-from-ocf": [
1325             "for x=0;x<numlinks(oic.r.heatingzonecollection): heatinglevels[x] = ocf.heatinglevel"
1326         ]
1327     }
1328 }
1329 }
1330 }
1331 },
1332 "type": "object",
1333 "allOf": [
1334     {"$ref": "#/definitions/asa.operation.heatingzone"}
1335 ],
1336 "required": [ "numberofheatingzones", "maxheatinglevels", "heatinglevels" ]
1337 }
1338 }

```

1339 7.14 HVAC Fan Mode

1340 7.14.1 Derived model

1341 The derived model: "asa.operation.hvacfanmode".

1342 7.14.2 Property definition

1343 Table 30 provides the detailed per Property mapping for "asa.operation.hvacfanmode".

1344 **Table 30 – The property mapping for "asa.operation.hvacfanmode".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
mode	oic.r.mode	modearray = [Auto,Circulation,Continuous]ocf.mode[0] = modearray[mode]	modearray = [Auto,Circulation,Continuous]mode = indexof modeArray[ocf.mode[0]]
supportedmodes	oic.r.mode	modearray = [Auto,Circulation,Continuous]for x=0, x < sizeof(supportedmodes): ocf.supportedmodes[x] = modearray[supportedmodes[x]]	modearray = [Auto,Circulation,Continuous]for x=0, x < sizeof(supportedmodes): supportedmodes[x] = indexof modearray[ocf.supportedmodes[x]]

1345 Table 31 provides the details of the Properties that are part of "asa.operation.hvacfanmode".

Table 31 – The properties of "asa.operation.hvacfanmode".

AllJoyn name	Property	Type	Required	Description
mode		integer	yes	Current mode of device.
supportedmodes		array	yes	Array of supported modes

1347 **7.14.3 Derived model definition**

```

1348 {
1349   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.hvacfanmode.json#",
1350   "$schema": "http://json-schema.org/draft-04/schema#",
1351   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1352   "title": "HVAC Fan Mode",
1353   "definitions": {
1354     "asa.operation.hvacfanmode": {
1355       "type": "object",
1356       "properties": {
1357         "mode": {
1358           "type": "integer",
1359           "description": "Current mode of device.",
1360           "x-ocf-conversion": {
1361             "x-ocf-alias": "oic.r.mode",
1362             "x-to-ocf": [
1363               "modearray = [Auto,Circulation,Continuous]",
1364               "ocf.mode[0] = modearray[mode]"
1365             ],
1366             "x-from-ocf": [
1367               "modearray = [Auto,Circulation,Continuous]",
1368               "mode = indexof modeArray[ocf.mode[0]]"
1369             ]
1370           }
1371         },
1372         "supportedmodes": {
1373           "type": "array",
1374           "items": {
1375             "type": "integer"
1376           },
1377           "description": "Array of supported modes",
1378           "x-ocf-conversion": {
1379             "x-ocf-alias": "oic.r.mode",
1380             "x-to-ocf": [
1381               "modearray = [Auto,Circulation,Continuous]",
1382               "for x=0, x < sizeof(supportedmodes): ocf.supportedmodes[x] =
1383 modearray[supportedmodes[x]]"
1384             ],
1385             "x-from-ocf": [
1386               "modearray = [Auto,Circulation,Continuous]",
1387               "for x=0, x < sizeof(supportedmodes): supportedmodes[x] = indexof
1388 modearray[ocf.supportedmodes[x]]"
1389             ]
1390           }
1391         }
1392       }
1393     }
1394   },
1395   "type": "object",
1396   "allOf": [
1397     { "$ref": "#/definitions/asa.operation.hvacfanmode" }
1398   ],
1399   "required": [ "mode", "supportedmodes" ]
1400 }
1401

```

1402 **7.15 On/Off Control**

1403 **7.15.1 Derived model**

1404 The derived model: "asa.operation.offcontrol".

1405 The derived model: "asa.operation.oncontrol".

1406 **7.15.2 Property definition**

1407 Table 32 provides the detailed per Property mapping for "asa.operation.offcontrol".

1408 **Table 32 – The property mapping for "asa.operation.offcontrol".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
switchon	oic.r.switch.binary	value = false	if ocf.value = false, asa.operation.offcontrol::switchoff().

1409 Table 33 provides the details of the Properties that are part of "asa.operation.offcontrol".

1410 **Table 33 – The properties of "asa.operation.offcontrol".**

AllJoyn name	Property	Type	Required	Description
switchon		string	no	Turn off the device

1411 Table 34 provides the detailed per Property mapping for "asa.operation.oncontrol".

1412 **Table 34 – The property mapping for "asa.operation.oncontrol".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
switchon	oic.r.switch.binary	value = true	if ocf.value = true, asa.operation.oncontrol::switchon().

1413 Table 35 provides the details of the Properties that are part of "asa.operation.oncontrol".

1414 **Table 35 – The properties of "asa.operation.oncontrol".**

AllJoyn name	Property	Type	Required	Description
switchon		string	no	Turn on the device

1415 **7.15.3 Derived model definition**

```

1416 {
1417   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.oncontrol.json#",
1418   "$schema": "http://json-schema.org/draft-04/schema#",
1419   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1420   "title": "On/Off Control",
1421   "definitions": {
1422     "asa.operation.oncontrol": {
1423       "type": "object",
1424       "properties": {
1425         "switchon": {
1426           "type": "string",
1427           "format": "method",
1428           "description": "Turn on the device",
1429           "x-ocf-conversion": {
1430             "x-ocf-alias": "oic.r.switch.binary",
1431             "x-to-ocf": [
1432               "value = true"
1433             ],
1434             "x-from-ocf": [
1435               "if ocf.value = true, asa.operation.oncontrol::switchon()."
1436             ]
1437           }
1438         }
1439       }
1440     }
1441   }

```

```

1437     }
1438   }
1439 }
1440 },
1441 "asa.operation.offcontrol": {
1442   "type": "object",
1443   "properties": {
1444     "switchon": {
1445       "type": "string",
1446       "format": "method",
1447       "description": "Turn off the device",
1448       "x-ocf-conversion": {
1449         "x-ocf-alias": "oic.r.switch.binary",
1450         "x-to-ocf": [
1451           "value = false"
1452         ],
1453         "x-from-ocf": [
1454           "if ocf.value = false, asa.operation.offcontrol::switchoff()."
1455         ]
1456       }
1457     }
1458   }
1459 },
1460 },
1461 "type": "object",
1462 "oneOf": [
1463   {"$ref": "#/definitions/asa.operation.oncontrol"},
1464   {"$ref": "#/definitions/asa.operation.offcontrol"}
1465 ]
1466 }
1467

```

1468 7.16 On Off Mapping

1469 7.16.1 Derived model

1470 The derived model: "asa.operation.onoffstatus".

1471 7.16.2 Property definition

1472 Table 36 provides the detailed per Property mapping for "asa.operation.onoffstatus".

1473 **Table 36 – The property mapping for "asa.operation.onoffstatus".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
onoff		oic.r.switch.binary	value = onoff	onoff = value

1474 Table 37 provides the details of the Properties that are part of "asa.operation.onoffstatus".

1475 **Table 37 – The properties of "asa.operation.onoffstatus".**

AllJoyn name	Property	Type	Required	Description
onoff		boolean	yes	On/Off status of the device

1476 7.16.3 Derived model definition

```

1477 {
1478   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.onoffstatus.json#",
1479   "$schema": "http://json-schema.org/draft-04/schema#",
1480   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1481   "title": "On Off Mapping",
1482   "definitions": {
1483     "asa.operation.onoffstatus": {
1484       "type": "object",
1485       "properties": {
1486         "onoff": {
1487           "type": "boolean",

```



```

1488     "description": "On/Off status of the device",
1489     "x-ocf-conversion": {
1490       "x-ocf-alias": "oic.r.switch.binary",
1491       "x-to-ocf": [
1492         "value = onoff"
1493       ],
1494       "x-from-ocf": [
1495         "onoff = value"
1496       ]
1497     }
1498   }
1499 }
1500 }
1501 },
1502 "type": "object",
1503 "allOf": [
1504   {"$ref": "#/definitions/asa.operation.onoffstatus"}
1505 ],
1506 "required": [ "onoff" ]
1507 }
1508

```

1509 **7.17 Oven Cycle Phase**

1510 **7.17.1 Derived model**

1511 The derived model: "asa.operation.ovencyclephase".

1512 **7.17.2 Property definition**

1513 Table 38 provides the detailed per Property mapping for "asa.operation.ovencyclephase".

1514 **Table 38 – The property mapping for "asa.operation.ovencyclephase".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
getvendorphasesdescription	oic.r.action		
supportedcyclephases	oic.r.operationalstate	phasearray = [Unavailable,Preheating,Cooking,Cleaning]for x=0, x < sizeof(supportedcyclephases): machinestates[x] = phasearray[supportedcyclephases[x]]	phasearray = [Unavailable,Preheating,Cooking,Cleaning]for x=0, x < sizeof(machinestates): supportedcyclephases[x] = indexof phasearray[machinestates[x]]
cyclephase	oic.r.operationalstate	phasearray = [Unavailable,Preheating,Cooking,Cleaning]currentmachinestate = phasearray[cyclephase]	phasearray = [Unavailable,Preheating,Cooking,Cleaning]cyclephase = indexof statearray[currentmachinestate[0]]

1515 Table 39 provides the details of the Properties that are part of "asa.operation.ovencyclephase".

1516 **Table 39 – The properties of "asa.operation.ovencyclephase".**

AllJoyn Property name	Type	Required	Description
getvendorphasesdescription		no	Get cycle phases description
supportedcyclephases	array	yes	Array of cycle phases supported by the Appliance.

cyclephase	integer	yes	Current phase of the operational cycle
------------	---------	-----	--

1517 **7.17.3 Derived model definition**

```

1518 {
1519   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.ovencyclephase.json#",
1520   "$schema": "http://json-schema.org/draft-04/schema#",
1521   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1522   "title": "Oven Cycle Phase",
1523   "definitions": {
1524     "asa.operation.ovencyclephase": {
1525       "type": "object",
1526       "properties": {
1527         "cyclephase": {
1528           "type": "integer",
1529           "description": "Current phase of the operational cycle",
1530           "x-ocf-conversion": {
1531             "x-ocf-alias": "oic.r.operationalstate",
1532             "x-to-ocf": [
1533               "phasearray = [Unavailable,Preheating,Cooking,Cleaning]",
1534               "currentmachinestate = phasearray[cyclephase]"
1535             ],
1536             "x-from-ocf": [
1537               "phasearray = [Unavailable,Preheating,Cooking,Cleaning]",
1538               "cyclephase = indexOf statearray[currentmachinestate[0]]"
1539             ]
1540           }
1541         },
1542         "supportedcyclephases": {
1543           "type": "array",
1544           "items": {
1545             "type": "integer"
1546           },
1547           "description": "Array of cycle phases supported by the Appliance.",
1548           "x-ocf-conversion": {
1549             "x-ocf-alias": "oic.r.operationalstate",
1550             "x-to-ocf": [
1551               "phasearray = [Unavailable,Preheating,Cooking,Cleaning]",
1552               "for x=0, x < sizeof(supportedcyclephases): machinestates[x] =
1553 phasearray[supportedcyclephases[x]]"
1554             ],
1555             "x-from-ocf": [
1556               "phasearray = [Unavailable,Preheating,Cooking,Cleaning]",
1557               "for x=0, x < sizeof(machinestates): supportedcyclephases[x] = indexOf
1558 phasearray[machinestates[x]]"
1559             ]
1560           }
1561         },
1562         "getvendorphasesdescription": {
1563           "x-ocf-type": "method",
1564           "description": "Get cycle phases description",
1565           "x-ocf-conversion": {
1566             "x-ocf-alias": "oic.r.action"
1567           }
1568         }
1569       }
1570     }
1571   },
1572   "type": "object",
1573   "allOf": [
1574     {"$ref": "#/definitions/asa.operation.ovencyclephase"}
1575   ],
1576   "required": [ "cyclephase", "supportedcyclephases" ]
1577 }
1578

```